**APPLICATION ASSESSMENT**

**Executive Summary:**

This report provides an overview of the approach and results used for building a Machine learning Model to predict fraud and save friction to legitimate customers.

**Approach:**

***Data Preparation:***The dataset consisted of historical transaction data having multiple features both categorical and numerical with time stamp information. Data preprocessing involved handling missing values, scaling numerical features, and encoding categorical variables. I used 70% data to train the Model and 30% data to test the model. (symmetric split – ratio of Fraud vs non-Fraud data was maintained in test and train data)

***Model Selection:***I have chosen XGboost classifier (SOTA ML decision tree model) out of 7 ML models i.e, logistic regression, decision tree, random forest, LightGBM, CatBoost, sklearn\_histGB and sklearnGB. The Model performs exceptionally well on unseen data with the ability to identify the **Fraudulent transaction up to ~75% of the time** and the **friction of the customer is around 1%**, which translates to 99 % of the time a wrong customer would not be squashed by the system. To take care of the trade-off between reducing Friction to customers and not missing frauds I made the classification threshold to 0.4 (default = 0.5). The F1 score of the final model is 80%.

**Limitations:**

1. There is huge number of categories in device\_info\_2 and the list does not look exhaustive. If this happens, the model will explode with number of columns to fit the model and produce ineffective results -> [Suggested solution: Plan to group some of the least frequent devices or bin few according to same category]
2. Feature Engineering: As most of the columns are very closely related, Combining and removing columns will make the prediction faster (as of now they are masked, removing columns might make the predictions weaker and unreliable).
3. As there are still 1% customer affected by the friction, another verification step like sending email/SMS to customer stating the fraudulent action.

**Recommendations and Conclusion:**  
As the trend of the fraud data keeps changing over period of time, we have to make sure to retrain the model in weekly to Bi-weekly basis (Champion-Challenger model). We can use Deep Learning model, if we have more data points.

**Bonus 1:**

Inference pipeline of XGboost classifier Model is deployed using Flask API. We need to incorporate model updates into the CI/CD pipeline (Jenkins, GitLab) to automate the process of model training, evaluation, and deployment to minimize downtime.Using version control systems to track changes to Fraud detection model code, data, and configuration. This helps in keeping a history of model versions.

**Bonus 2:**

I used K-means and Isolation Forest for detecting Fraud. K-means is not doing a great job in clustering transactions. Isolation Forest did a decent job comparatively. If there are more data and clear decision boundary, we can incorporate semi-supervised models like autoencoders.